```
*  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
FILE 'USPAT' ENTERED AT 15:13:58 ON 04 NOV 1997

    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
    *                   W E L C O M E   T O   T H E                     *
    *           U. S.   P A T E N T   T E X T   F I L E                 *
    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
```

=> s 5347653/uref
L1   14 5347653/UREF

=> d bro
:1

1. 5,685,003, Nov. 4, 1997, Method and system for automatically indexing
data in a document using a fresh index table; Kyle Gordon Peltonen, et
al., 1/1 :IMAGE AVAILABLE:
:ab

US PAT NO: 5,685,003 :IMAGE AVAILABLE: L1: 1 of 14

ABSTRACT:
A system and method for indexing words in documents, the system including
a master index for storing the words and for storing associated index
data. One of the documents is selected for updating and is edited. Next,
a shadow index is created. Each word from the selected edited document is
then indexed in the shadow index. A fresh index table is updated to
indicate that the shadow index contains the most up-to-date data
regarding the selected edited document. Query requests will be processed
by searching all relevant indexes and comparing the retrieved results
with the data in the fresh index table. Only the most up-to-date data
will actually be returned as the query results. Periodically, shadow
indexes and the master index can be merged into a new master index. Only
the most up-to-date data, as determined by a comparison with the fresh
index table, will be stored in the new master index.
:2

2. 5,684,991, Nov. 4, 1997, Modification metadata set, abstracted from
database write requests; Peter Bryan Malcolm, 395/182.04 :IMAGE
AVAILABLE:
:ab

US PAT NO: 5,684,991 :IMAGE AVAILABLE: L1: 2 of 14

ABSTRACT:
In a microprocessor-based computer system, file backups are performed by
monitoring modified areas of one or more files, through the write
statements which give rise to the modifications. A meta-data database is
constructed containing information relating to the locations of those
modified areas. The backup database may then be modified by interrogating
the meta-data database and thereby backing up only the data which has
been modified in the database itself. Thus the size of the backup
database may be reduced. The invention may be incorporated in a dedicated
backup application and/or operating system or may interact with a
conventional file-by-file backup application.
:3

3. 5,680,621, Oct. 21, 1997, System and method for domained incremental
changes storage and retrieval; Roni Korenshtein, 395/705 :IMAGE

4.   5,680,613, Oct. 21, 1997, Data processing system using versioned documents having temporary links; Ryo Atsumi, 395/701 :IMAGE AVAILABLE:

5.   5,671,398, Sep. 23, 1997, Method for collapsing a version tree which depicts a history of system data and processes for an enterprise; Ronald Jay Neubauer, 395/500, 712 :IMAGE AVAILABLE:

6.   5,668,991, Sep. 16, 1997, Database management system; Stephen Dunn, et al., 707/202 :IMAGE AVAILABLE:

7.   5,638,509, Jun. 10, 1997, Data storage and protection system; William E. Dunphy, et al., 395/182.18 :IMAGE AVAILABLE:

8.   5,623,662, Apr. 22, 1997, Revenue sharing system with data filtering using history, periodic, and exclusion databases; Matthew S. McIntosh, 707/204; 364/282.1, DIG.1 :IMAGE AVAILABLE:

9.   5,592,661, Jan. 7, 1997, Detection of independent changes via change identifiers in a versioned database management system; Neal R. Eisenberg, et al., 707/104; 364/282.1, 282.3, 283.1, DIG.1 :IMAGE AVAILABLE:

10.   5,590,318, Dec. 31, 1996, Method and system for tracking files pending processing; Mark Zbikowski, et al., 707/202; 395/182.13, 182.17, 182.18 :IMAGE AVAILABLE:

11.   5,581,755, Dec. 3, 1996, Method for maintaining a history of system data and processes for an enterprise; Paul D. Koerber, et al., 707/103; 364/268, 964.34, DIG.1, DIG.2; 395/670, 701 :IMAGE AVAILABLE:

12.   5,574,906, Nov. 12, 1996, System and method for reducing storage requirement in backup subsystems utilizing segmented compression and differencing; Robert J. T. Morris, 707/1; 364/222.82, 282.1, 285.1, 962.1, 974, DIG.1, DIG.2 :IMAGE AVAILABLE:

13.   5,559,991, Sep. 24, 1996, Incremental computer file backup using check words; Arnon Kanfi, 711/162; 395/182.11; 707/203 :IMAGE AVAILABLE:

14.   5,495,600, Feb. 27, 1996, Conversion of queries to monotonically increasing incremental form to continuously query a append only database; Douglas B. Terry, et al., 707/3; 364/974, 974.6, DIG.2 :IMAGE AVAILABLE: :3-14 ab

US PAT NO: 5,680,621 :IMAGE AVAILABLE: L1: 3 of 14

ABSTRACT:
A computer or computer program product having a computer-readable medium comprising at least one controller to enable a processor to perform incremental changes storage and retrieval. The at least one controller comprises an incremental storage adaptor for enabling the processor to store a change associated with a base file, rather than modifying the base file by the change and storing the result as a new file. An incremental retrieval adaptor is included for enabling the processor to merge the change with the base file thereby forming a logical new file. The incremental retrieval adaptor can perform the merging in response to a call from an application running on the processor or another processor, and the incremental retrieval adaptor then provides or otherwise makes the logical new file available to the application.

US PAT NO: 5,680,613 :IMAGE AVAILABLE: L1: 4 of 14

ABSTRACT:
The processing operations of an actual desk work can be executed as they

are, and the efficiency of data processing work can be significantly improved by utilizing the p●●ssing capability of a compute●●ystem. A data processing system utili●●ng computer processing a means ●r generating a versioned documents which is confined and comprehesive type of tables/lists for maintaining initial data in their original state as they are, a means for generating a ledger which is a table/list including information which vary and change with time, and a means for generating a document control register which is a kind of ledger for maintaining the identification of tables/lists for controlling. The above-mentioned document control register has a portion for identifying and controlling its own records and tables/lists which are the object of control and a portion for controlling their conditions for use and states so that all document data can be included, controlled, and maintained by this document control register. In particular, the versioned documents maintain temporary links between the objects of control.

US PAT NO: 5,671,398 :IMAGE AVAILABLE: L1: 5 of 14

ABSTRACT:
The method of the present invention is useful in a computer system having a user interface, a memory, a database and a repository operative in the computer system for accessing the database. The method is implemented by the computer system for collapsing a version tree that depicts a history of objects stored in the database. The method, which is stored in the memory at run-time, comprises the steps of validating collapse request by insuring that the target object is not a ghost object, the end object is not a ghost, the end object is on the same version tree as the target object, and the end object is a later version than the target object. An empty array representing path objects is initialized from the target object to the end object; and if an end object is supplied, a function is called to build up an array of objects representing paths from the target object to the end object. The array of objects built up in the preceding steps is processed. For each nextVersion of the target object, a function is called to collapse the versions starting from nextVersion; and, a function is also called to destruct the target object.

US PAT NO: 5,668,991 :IMAGE AVAILABLE: L1: 6 of 14

ABSTRACT:
A database system is described in which changes to a database file are recorded in a journal file. The journal file comprises a sequence of after-image records each having a database identifier indicating which database record it relates. Periodically, a dump is taken of the database. At regular intervals, between dumps the journal file is merged with any previously archived journal file to form a merged file, the records are sorted into order of their database identifiers and the merged file is filtered to remove all but the latest after-image record relating to each particular database record. The filtered journal file is archived. If the database file is corrupted, it can be recovered by applying the filtered journal file to a dump file. Because the journal file has been filtered, recovery is faster. The filtered journal can also be used for performing an off-line integrity check on the reconstructed database file.

US PAT NO: 5,638,509 :IMAGE AVAILABLE: L1: 7 of 14

ABSTRACT:
The data storage and protection apparatus maintains an index of all data file activity on a computer system and stores copies of data files in a manner to enable a user to recreate the state of the computer system at any selected point in time. A data file monitor intercepts all communication between application programs and the file system to obtain data file status and activity information. This information is used to identify data files which are transmitted to a data file backup media for storage along with directory information that identifies the stored files

to enable the system to later locate and retrieve data files that were stored on the backup media.

ABSTRACT:
A method and system for extracting revenue information from a
point-of-sale (POS) terminal for purposes of revenue sharing including
the step of periodically selecting and extracting predetermined portions
of revenue transaction data in a proprietary database. This system allows
extrapolation of select data relating to revenue traffic in a rental
system. Revenue stored in a proprietary database (40) by a proprietary
point-of-sale operating program is periodically selected, extracted and
stored in a periodic database (42). This data is compared to a history
database (38) comprising the periodic database as it existed prior to a
given time. The data is output from the periodic database that is not
present in the history database at the given time. This data can be
transmitted to a remote location. The proprietary point-of-sale operating
program can be used to create a history report database from the revenue
transaction data and the portions of the revenue transaction data can be
selected and extracted from the history report database.

ABSTRACT:
A versioned-data management system is provided with a method for
detecting independent changes to a part when an attempt is made to
promote a variant of the part. For each variant of a part, there is
maintained two identifiers and a flag. The first identifier is the
identifier of the last change to that variant. The second identifier is
the last-change identifier of the variant which was the source of the
drawdown. When the variant is successfully promoted, and there was not
previously a variant at the level to which it is promoted, its
drawdown-from identifier value is not changed. When the variant is
successfully promoted, and there was previously a variant at the level to
which it is promoted, the drawdown-from identifier is set to the
drawdown-from identifier value of the variant which was previously at the
level to which the promote is made. When the promote of a variant fails,
the drawdown-from identifier is set to the last-change identifier value
of the variant which is first in the search path beginning at the level
to which the promote is being made. The flag indicates whether the
variant can be promoted by normal means, or must be promoted using
promote.sub.-- force. The identifiers and flag are used to identify
independent changes and control data integrity in the VDMS at promotion
time.

ABSTRACT:
A portion of a first storage stores identifying information of files to
be processed by a computer. The files are stored in a second storage,
with each file having associated identifying information which identifies
the file, and each file including an update sequence value which
indicates an order in which the file is to be processed by the computer.
A threshold value is provided which is set equal to a largest update
sequence value retrieved from the file that have been processed by the
computer. In response to a loss of the file identifiers stored in the
portion of the first storage, the file identifier for each file whose
update sequence value is beyond the threshold value is stored in the
portion of the first storage.

ABSTRACT:
The method of the present invention is useful in a computer system having

a user interface, a memory, a repository and a database. The method is a repository program executed ● the computer system for maint●ing a history of objects stored in the repository. The method compr●es the steps of determining if the object is a new logical object, and if so constructing a new versioned object; and, if not retrieving from the repository an object on which a new state is to be based. If the new state is being derived from only one object, then reserving the object. If the state is being derived from two objects, then retrieving from the repository the other object on which the new state is based, and merging the two objects. If the state changes are to be retained, then replacing the object; and, if the state changes are not to be retained, then unreserving the object.

US PAT NO: 5,574,906 :IMAGE AVAILABLE: L1: 12 of 14

ABSTRACT:
In a client/server environment, a method and means for reducing the storage requirement in the backup subsystem and further reducing the load on the transmission bandwidth where base files are maintained on the server in a segmented compressed format. When a file is modified on the client, the file is transmitted to the server and compared with the segmented compressed base version of the file utilizing a differencing function but without decompressing the entire base file. A delta file which is the difference between the compressed base file and the modified version of the file is created and stored on a storage medium which is part of the backup subsystem. Alternatively, a copy of frequently accessed base files are maintained on the client in a compressed format. Whenever the client detects that a frequently accessed file has been modified, the modified version of the file is differenced against the base version of that file without decompressing the entire base file and a delta file is generated. The delta file is then transmitted to the server to be stored at the server for storage medium to be utilized either immediately or at a later time to update the base version of the modified file on the server.

US PAT NO: 5,559,991 :IMAGE AVAILABLE: L1: 13 of 14

ABSTRACT:
A facility is provided for storing in a backup memory only those blocks of a file, or disk partition, which differ from corresponding blocks forming an earlier version of the file. Specifically, a file is divided into a number of blocks and a "signature" is generated for each such block. A block is then stored in the backup memory only if its associated signature differs from a signature generated for an earlier version of the block. In addition, if two blocks of the current version of the file have identical signatures and are to be stored in the backup memory, then only one of the two blocks is stored in the memory and a simple message indicating that the other block is equal to the one block is stored in the memory for the other block. Further, the application of such signatures is advantageously applied to the opposite case of restoring a file using copies of previous versions of the file that are stored in the backup memory.

US PAT NO: 5,495,600 :IMAGE AVAILABLE: L1: 14 of 14

ABSTRACT:
To produce a continuous query for an append-only database, a client defined query first is converted into its minimal bounding non-decreasing monotone (hereinafter referred to as "monotonic increasing" query. This monotonic query, in turn, is converted into an incremental query, The resulting monotonically increasing incremental query then is installed on the database as a stored procedure that takes two date/time parameters (hereinafter referred to as "time" parameters), one of which (.tau.) identifies the last time the procedure was executed, and the other of which (t) identifies the current time. All database records are

timestamped as of the time that they are entered into the database. Thus, in operation, more or less ●ndard procedure calls periodic●y invoke each of the stored query pro●dures, thereby periodically exe●ting the incremental queries over database records that have timestamps spanning successive time slots.
s

```
=> homer?/xa
'HOMER?' IS NOT A RECOGNIZED COMMAND

=> s homer?/xa
L2          125 HOMER?/XA

=> d bro
:1

1.  5,684,989, Nov. 4, 1997, Terminal database managing system as latest
version; Hiroyuki Nissato, 395/839, 853 :IMAGE AVAILABLE:
:2

2.  5,682,497, Oct. 28, 1997, Managing file structures for a flash memory
file system in a computer; Kurt B. Robinson, 1/1 :IMAGE AVAILABLE:
:3

3.  5,680,605, Oct. 21, 1997, Method and apparatus for searching a large
volume of data with a pointer-based device in a data processing system;
Robert J. Torres, 1/1 :IMAGE AVAILABLE:
:end

=> s ((delta por differ?) and (updat? or version?))/ab
         4870 DELTA/AB
           39 POR/AB
       160046 DIFFER?/AB
            0 DELTA POR DIFFER?/AB
               ((DELTA(W)POR(W)DIFFER?)/AB)
         4860 UPDAT?/AB
         3189 VERSION?/AB
L3          0 ((DELTA POR DIFFER?) AND (UPDAT? OR VERSION?))/AB

=>s ((delta# or differ?) and (updat? or version?))/ab
         4873 DELTA#/AB
       160046 DIFFER?/AB
         4860 UPDAT?/AB
         3189 VERSION?/AB
L4       1340 ((DELTA# OR DIFFER?) AND (UPDAT? OR VERSION?))/AB

=> s l4 and (search? or retriev?)/ab
         3770 SEARCH?/AB
         7796 RETRIEV?/AB
L5         60 L4 AND (SEARCH? OR RETRIEV?)/AB

=> s l5 and (match? or compar?)/ab
        17746 MATCH?/AB
        76564 COMPAR?/AB
L6         25 L5 AND (MATCH? OR COMPAR?)/AB

=> d bro
:1

1.  5,642,502, Jun. 24, 1997, Method and system for searching for
relevant documents from a text database collection, using statistical
ranking, relevancy feedback and small pieces of text; James R. Driscoll,
707/5; 364/225.4, 274.8, 282.1, DIG.1 :IMAGE AVAILABLE:
:xa
```

2. 5,634,124, May 27, 1997, Data integration by object management; Dana Khoyi, et al., 707/103; 395/683 :IMAGE AVAILABLE:
:ab

ABSTRACT:
An object based data processing system including an extensible set of object types and a corresponding set of "object managers" wherein each object manager is a program for operating with the data stored in a corresponding type of object. The object managers in general support at least a standard set of operations. Any program can effect performance of these standard operations on objects of any type by making an "invocation" request. In response to an invocation request, object management services (which are available to all object managers) identifies and invokes an object manager that is suitable for performing the requested operation on the specified type of data. A mechanism is provided for linking data from one object into another object. A object catalog includes both information about objects and about links between objects. Data interchange services are provided for communicating data between objects of **different** types, using a set of standard data interchange formats. A **matchmaker** facility permits two processes that are to cooperate in a data interchange operation identify each other and to identify data formats they have in common. A facility is provided for managing shared data "resources", Customized **versions** of resources can be created and co-exist with standard resources. A resource **retrieval** function determines whether a customized or a standard resource is to be returned in response to each request for a resource.
:3

3. 5,598,354, Jan. 28, 1997, Motion video compression system with neural network having winner-take-all function; Wai-Chi Fang, et al., 364/715.02 :IMAGE AVAILABLE:
:4

4. 5,533,179, Jul. 2, 1996, Apparatus and method of modifying hardware description language statements; Kayhan Kucukcakar, et al., 345/433 :IMAGE AVAILABLE:
:5

5. 5,490,201, Feb. 6, 1996, Method in conjunction with updating one or more HLR-databases included in a mobile telephone system; Jorgen Moberg, et al., 455/433 :IMAGE AVAILABLE:
:6

6. 5,486,864, Jan. 23, 1996, Differential time code method and apparatus as for a compressed video signal; Joel W. Zdepski, 348/423, 500 :IMAGE AVAILABLE:
:6

6. 5,486,864, Jan. 23, 1996, Differential time code method and apparatus as for a compressed video signal; Joel W. Zdepski, 348/423, 500 :IMAGE AVAILABLE:
:7

7. 5,481,713, Jan. 2, 1996, Method and apparatus for patching code residing on a read only memory device; Russ Wetmore, et al., 395/705, 712 :IMAGE AVAILABLE:
:8

8. 5,467,137, Nov. 14, 1995, Method and apparatus for synchronizing a

receiver as for a compressed video signal using differential time code;
Joel W. Zdepski, 348/845.3, ⬤0 :IMAGE AVAILABLE: ⬤
:9

9. 5,421,015, May 30, 1995, Multitasking system having an application
integration mechanism for linking differently typed data objects; Dana
Khoyi, et al., 395/677; 364/DIG.1; 395/683, 685 :IMAGE AVAILABLE:
:ab

ABSTRACT:
An object based data processing system including an extensible set of
object types and a corresponding set of "object managers" wherein each
object manager is a program for operating with the data stored in a
corresponding type of object. The object managers in general support at
least a standard set of operations. Any program can effect performance of
these standard operations on objects of any type by making an
"invocation" request. In response to an invocation request, object
management services (which are available to all object managers)
identifies and invokes an object manager that is suitable for performing
the requested operation on the specified type of data. A mechanism is
provided for linking data from one object into another object. A object
catalog includes both information about objects and about links between
objects. Data interchange services are provided for communicating data
between objects of **different** types, using a set of standard data
interchange formats. A **matchmaker** facility permits two processes that
are to cooperate in a data interchange operation identify each other and
to identify data formats they have in common. A facility is provided, for
managing shared data "resources". Customized **versions** of resources
can be created and co-exist with standard resources. A resource
retrieval function determines whether a customized or a standard
resource is to be returned in response to each request for a resource.
:11

11. 5,349,651, Sep. 20, 1994, System for translation of virtual to
physical addresses by operating memory management processor for
calculating location of physical address in memory concurrently with
cache comparing virtual addresses for translation; Ricky C. Hetherington,
et al., 711/207; 364/243.41, 256.3, 256.4, 256.5, 961.2, 961.4, DIG.1,
DIG.2 :IMAGE AVAILABLE:
:12

12. 5,303,379, Apr. 12, 1994, Link mechanism for linking data between
objects and for performing operations on the linked data in an object
based system; Dana Khoyi, et al., 395/710; 364/280, 284.3, DIG.1;
395/670, 683 :IMAGE AVAILABLE:
:13

13. 5,261,080, Nov. 9, 1993, Matchmaker for assisting and executing the
providing and conversion of data between objects in a data processing
system storing data in typed objects having different data formats; Dana
Khoyi, et al., 395/500; 364/260, 260.2, 260.3, DIG.1 :IMAGE AVAILABLE:
:ab

ABSTRACT:
An object based data processing system including an extensible set of
object types and a corresponding set of "object managers" wherein each
object manager is a program for operating with the data stored in a
corresponding type of object. The object managers in general support at
least a standard set of operations. Any program can effect performance of
these standard operations on objects of any type by making an
"invocation" request. In response to an invocation request, object

management services (which are available to all object managers)
identifies and invokes an o█ct manager that is suitable fo█erforming
the requested operation on t█ specified type of data. A mec█ism is
provided for linking data from one object into another object. A object
catalog includes both information about objects and about links between
objects. Data interchange services are provided for communicating data
between objects of **different** types, using a set of standard data
interchange formats. A **matchmaker** facility permits two processes that
are to cooperate in a data interchange operation identify each other and
to identify data formats they have in common. A facility is provided for
managing shared data "resources". Customized **versions** of resources
can be created and co-exist with standard resources. A resource
retrieval function determines whether a customized or a standard
resource is to be returned in response to each request for a resource.
:14

14. 5,252,965, Oct. 12, 1993, Changing one of many access codes upon
removal of ignition key; Sachal B. Gidwani, et al., 340/825.56, 825.31,
825.34 :IMAGE AVAILABLE:
:end

=> d his

        (FILE 'USPAT' ENTERED AT 15:13:58 ON 04 NOV 1997)
L1   14 S 5347653/UREF
L2   125 S HOMER?/XA
L3   0 S ((DELTA POR DIFFER?) AND (UPDAT? OR VERSION?))/AB
L4   1340 S ((DELTA# OR DIFFER?) AND (UPDAT? OR VERSION?))/AB
L5   60 S L4 AND (SEARCH? OR RETRIEV?)/AB
L6   25 S L5 AND (MATCH? OR COMPAR?)/AB

=> s 14 and (database# or data base# or dbms)/ab
          1968 DATABASE#/AB
         85647 DATA/AB
        159494 BASE#/AB
          1453 DATA BASE#/AB
                 ((DATA(W)BASE#)/AB)
            27 DBMS/AB
L7   76 L4 AND (DATABASE# OR DATA BASE# OR DBMS)/AB

=> s 15 and 17
L8   13 L5 AND L7

=> s ((delta# or differ?) and (updat? or version?) and (search? or retriev?)
and (database# or data base# or dbms))/clm
         14255 DELTA#/CLM
        355361 DIFFER?/CLM
         10753 UPDAT?/CLM
          3182 VERSION?/CLM
          6168 SEARCH?/CLM
         14323 RETRIEV?/CLM
          2949 DATABASE#/CLM
        122751 DATA/CLM
        347578 BASE#/CLM
          3581 DATA BASE#/CLM
                 ((DATA(W)BASE#)/CLM)
            18 DBMS/CLM
L9   112 ((DELTA# OR DIFFER?) AND (UPDAT? OR VERSION?) AND (SEARCH?
OR
               RETRIEV?) AND (DATABASE# OR DATA BASE# OR DBMS))/CLM

=> s 17 and 19
L10   11 L7 AND L9

=> d bro

1. 5,642,502, Jun. 24, 1997, Method and system for searching for
relevant documents from a text database collection, using statistical
ranking, relevancy feedback and small pieces of text; James R. Driscoll,
707/5; 364/225.4, 274.8, 282.1, DIG.1 :IMAGE AVAILABLE:
:2

2. 5,631,658, May 20, 1997, Method and apparatus for operating
geography-altering machinery relative to a work site; Adam J. Gudat, et
al., 342/457, 357; 701/1 :IMAGE AVAILABLE:
:3

3. 5,608,898, Mar. 4, 1997, Development system with methods for
maintaining data integrity of information stored as a data record in a
database with a copy of the information displayed on a screen device;
William M. Turpin, et al., 707/201; 364/282.1, 283.3, 283.4, DIG.1 :IMAGE
AVAILABLE:
:4

4. 5,572,424, Nov. 5, 1996, Diagnostic system for an engine employing
collection of exhaust gases; Robert L. Kellogg, et al., 701/29 :IMAGE
AVAILABLE:
:5

5. 5,471,391, Nov. 28, 1995, Method and apparatus for operating
compacting machinery relative to a work site; Adam J. Gudat, et al.,
701/50; 340/995; 364/151 :IMAGE AVAILABLE:
:6

6. 5,448,729, Sep. 5, 1995, Office system with audit history; David M.
Murdock, 707/104; 364/225.1, 225.3, 225.4, 282.1, 283.1, DIG.1; 395/180
:IMAGE AVAILABLE:
:7

7. 5,442,765, Aug. 15, 1995, Database system which adjusts the data
storage order based on the processing speed of the storage media; Shoji
Shiga, 707/8; 364/964.2, 964.6, 969.2, DIG.2; 711/117 :IMAGE AVAILABLE:
:8

8. 5,347,653, Sep. 13, 1994, System for reconstructing prior versions of
indexes using records indicating changes between successive versions of
the indexes; Rex A. Flynn, et al., 707/203; 364/255.2, 282.1, 285.1,
DIG.1; 711/162 :IMAGE AVAILABLE:
:ab

US PAT NO: 5,347,653 :IMAGE AVAILABLE: L10: 8 of 11

ABSTRACT:
A method and apparatus for providing an historical perspective into a
database of information objects through an efficient method and
apparatus for **versioning** information objects stored in a **database**
as well as an index representative of the information objects is
disclosed. The latest **versions** of each one of the information objects
and each one of the entries in the index are maintained in the
database. Partial earlier **versions** of each one of the information
objects and each one of the index entries are stored with the latest
versions. The partial **versions** contain only sufficient
information about the **differences** between the earlier **versions**
and the later ones so that any earlier **version** may be reconstructed.
Identification tags are employed as unique identifiers for each one of
the plurality of information objects and for **different** **versions**
of the information objects. Through the exploitation of the relative time
ordering of generated identification tags, an efficient index
representation and method for performing historical queries is provided.

9. 5,317,733, May 31, 1994, Office automation system for data base
management and forms generation; David M. Murdock, 707/203; 364/225.1,
225.3, 225.4, 282.1, 283.1, DIG.1 :IMAGE AVAILABLE:
:10

10. 4,847,807, Jul. 11, 1989, Multiple disk memory access arrangement
for gridded type data; Raymond C. Yule, 711/114; 364/925, 925.1, 933.8,
952, 952.1, 952.31, 962, 962.1, 964, 966.1, 966.4, DIG.2 :IMAGE
AVAILABLE:
:11

11. 4,745,559, May 17, 1988, Method and system for dynamically
controlling the content of a local receiver data base from a transmitted
data base in an information retrieval communication network; Richard A.
Willis, et al., 705/37; 370/428, 477; 395/200.47; 707/10 :IMAGE
AVAILABLE:
:end

=> d his full

        (FILE 'USPAT' ENTERED AT 15:13:58 ON 04 NOV 1997)
L1   14 SEA 5347653/UREF
L2   125 SEA HOMER?/XA
L3   0 SEA ((DELTA POR DIFFER?) AND (UPDAT? OR VERSION?))/AB
L4   1340 SEA ((DELTA# OR DIFFER?) AND (UPDAT? OR VERSION?))/AB
L5   60 SEA L4 AND (SEARCH? OR RETRIEV?)/AB
L6   25 SEA L5 AND (MATCH? OR COMPAR?)/AB
L7   76 SEA L4 AND (DATABASE# OR DATA BASE# OR DBMS)/AB
L8   13 SEA L5 AND L7
L9   112 SEA ((DELTA# OR DIFFER?) AND (UPDAT? OR VERSION?) AND (SEA
RCH
             ? OR RETRIEV?) AND (DATABASE# OR DATA BASE# OR DBMS))/CLM
L10   11 SEA L7 AND L9

        FILE USPAT
         *
         *   W E L C O M E   T O   T H E   *
         *   U. S.   P A T E N T   T E X T   F I L E   *
         *